OPAQUE: An Asymmetric PAKE Protocol Secure Against Pre-Computation Attacks *

Stanislaw Jarecki[1], Hugo Krawczyk[2], and Jiayu Xu[1]

[1] University of California, Irvine. Email: {stasio@ics.,jiayux@}uci.edu.
[2] IBM Research. Email: hugo@ee.technion.ac.il.

Abstract. Password-Authenticated Key Exchange (PAKE) protocols allow two parties that only share a password to establish a shared key in a way that is immune to offline attacks. *Asymmetric* PAKE (aPAKE) strengthens this notion for the more common client-server setting where the server stores a mapping of the password and security is required even upon server compromise; that is, the only allowed attack in this case is an (inevitable) offline exhaustive dictionary attack against individual user passwords. Unfortunately, current aPAKE protocols (that do not rely on PKI) allow for *pre-computation attacks* that lead to the *instantaneous compromise* of user passwords upon server compromise, thus forgoing much of the intended aPAKE security. Indeed, these protocols use – in essential ways – deterministic password mappings or use random "salt" transmitted *in the clear* from servers to users, and thus are vulnerable to pre-computation attacks.

We initiate the study of *Strong aPAKE* protocols that are secure as aPAKE's but *are also secure against pre-computation attacks.* We formalize this notion in the Universally Composable (UC) settings and present two modular constructions using an Oblivious PRF as a main tool. The first builds a Strong aPAKE from *any* aPAKE (which in turn can be constructed from any PAKE [26]) while the second builds a Strong aPAKE from *any* authenticated key-exchange protocol secure against KCI attacks. Using the latter transformation, we show *a practical instantiation of a UC-secure Strong aPAKE* in the Random Oracle model. The protocol ("OPAQUE") consists of 3 messages, requires 3 and 4 exponentiations for server and client, respectively (including a multi-exponentiation and 1 or 2 fixed-base per party), provides forward secrecy and explicit mutual authentication, is PKI-free, supports user-side password hardening, has a built-in facility for password-based storage-and-retrieval of secrets and credentials, and accommodates a user-transparent server-side threshold implementation.

1 Introduction

Passwords constitute the most ubiquitous form of authentication in the Internet, from the mundane to the most sensitive applications. The almost

* This is a revised ePrint version of the paper which appeared in Eurocrypt 2018 [33]. See revision notes in Sec. 1.2.